CLAIMS

1. A method for converting a file access data structure from a first endianness to a second endianness, the method comprising the steps of:

identifying, from a descriptor look up table, a series of actions to perform on elements of the file access data structure; and

performing the identified series of actions on the elements of the file access data structure.

2. A method of converting elements of a file access data structure from a first endianness to a second endianness, the method comprising the steps of:

determining if the file access data structure is a critical path data structure;

converting, in response to the file access data structure being a critical path data structure, the elements from the first endianness to the second endianness using a set of specific code functions;

converting, in response to the file access data structure not being a critical path data structure, a header of the file access data structure from the first endianness to the second endianness using a second set of specific code functions; and

calling a byte swapping engine to convert selected elements of the file access data structure from the first byte order to the second byte order.

3. The method of claim 2 wherein the file access data structure further comprises a direct access file access data structure.

4. A system for converting elements of a file access data structure from a first endianness to a second endianness, the system comprising:

an input buffer, the input buffer storing the file access data structure to be converted;

a byte swapping engine, the byte swapping engine operative interconnected with a descriptor table; and an output buffer, the byte swapping engine placing the file access data structure in the output buffer after conversion.

5. The system of claim 4 wherein the descriptor table further comprises a set of entries describing various file access data structures, each entry further comprising a size field and an operation field.

6. The system of claim 4 wherein the file access data structure further comprises a direct access file access data structure.

7. A method for converting a data structure from a first byte order to a second byte order, the method comprising the steps of:

reading an element entry from a descriptor table;

performing an action on an element of the data structure, the action being defined in the element entry read from the descriptor table; and

placing the element in an output buffer.

8. The method of claim 7 wherein the step of performing an action on an element further comprises the step of copying the element from an input buffer to the output buffer.

9. The method of claim 7 wherein the step of performing an action on an element further comprises the step of byte swapping the element.

10. The method of claim 7 wherein the element entry of the descriptor table further comprises a field describing a size of the element and a field describing an action to be performed.

11. A file server for use in a network storage environment, the file server comprising:

a byte swapping engine, the byte swapping engine performing a defined operation on each of a plurality of elements of a file access data structure.

12. The file server of claim 11 wherein the file server further comprises a descriptor look up table, the descriptor look up table having a plurality of entries, each of the plurality of entries associated with a specific file access data structure.

13. The file server of claim 12 wherein each of the plurality of entries further comprises a plurality of elements, each of the elements having a size field and an operation field.

14. The file server of claim 13 wherein the defined operation is defined by the operation field of the entry associated with the file access data structure.

15. A computer-readable medium, including program instructions executing on a computer, for converting elements of a file access data structure from a first endianness to a second endianness, the method comprising the steps of:

determining if the file access data structure is a critical path data structure;

converting, in response to the file access data structure being a critical path data structure, the elements from the first endianness to the second endianness using a set of specific code functions;

converting, in response to the file access data structure not being a critical path data structure, a header of the file access data structure from the first endianness to the second endianness using a second set of specific code functions; and

calling a byte swapping engine to convert selected elements of the file access data structure from the first byte order to the second byte order.

16. A method for converting elements of a file access data structure from a first endianness to a second endianness, the method comprising the steps of:

determining a type of the file access data structure;

processing, in response to the file access data structure of being of a first type, the file access data structure along a first processing path;

processing, in response to the file access data structure being of a second type, the file access data structure along a second processing path.

17. The method of claim 16 wherein the first type further comprises a critical path data structure.

18. The method of claim 16 wherein the first processing path further comprises a set of specifically coded functions.

19. The method of claim 16 wherein the second processing path further comprises a byte swapping engine.